

The determinism and boundedness of self-assembling structures

S. Tesoro¹ and S. E. Ahnert^{1,2}

¹*Theory of Condensed Matter, Cavendish Laboratory,
University of Cambridge, CB3 0HE Cambridge, UK*

²*Sainsbury Laboratory, University of Cambridge, CB2 1LR Cambridge, UK*

Self-assembly processes are widespread in nature, and lie at the heart of many biological and physical phenomena. The characteristics of self-assembly building blocks determine the structures that they form. Among the most important of these properties are whether the self-assembly is deterministic or nondeterministic, and whether it is bound or unbound. The former tells us whether the same set of building blocks always generates the same structure, and the latter whether it grows indefinitely. These properties are highly relevant in the context of protein structures, as the difference between deterministic protein self-assembly and nondeterministic protein aggregation is central to a number of diseases. Here we introduce a graph-based approach that can determine, with a few restrictions, whether a set of self-assembly building blocks is deterministic or nondeterministic, and whether it is bound or unbound. We apply this methodology to a previously studied lattice self-assembly model and discuss generalisations to a wide range of other self-assembling systems.

INTRODUCTION

Self-assembly is an ubiquitous phenomenon in nature, producing complex structures in biology, chemistry and physics. Examples include DNA [1–7], protein quaternary structure [8, 9], protein aggregation [10], viruses [11], micelles [12], and thin films [13]. Perhaps the two most fundamental properties of self-assembling systems are (a) whether the building blocks always form the same structure or not, and (b) whether the self-assembling structure grows indefinitely or remains bounded. The former property is sometimes referred to as the *determinism* and the latter as *boundedness* [14, 15]. Protein complexes are a prominent example of deterministic, bound self-assembly. Misfolding or erroneous binding of mutated versions of such proteins can cause the self-assembly of a protein complex to become nondeterministic, unbound protein aggregation, which in turn is the hallmark of a number of severe diseases, such as sickle-cell anaemia, and Alzheimer’s [10, 16]. In this paper we introduce a framework for establishing whether a given set of self-assembly building blocks is deterministic or nondeterministic, and whether it is bound or unbound. We apply this approach to a previously studied lattice self-assembly model introduced in [17] and studied further in [14, 15, 18–20]. In this model square tiles have attractive interfaces of different types. The interfaces can be symmetric or asymmetric, and the interface type is denoted by an integer (sometimes termed the ‘colour’ of the interface). The interactions between colours are specified by a binary interaction matrix, the assembly process is stochastic, often starting from a single seed tile, and the final structures assembled in this way represent sets of connected lattice sites that are sometimes referred to as Polyominoes. Such tile self-assembly or Polyomino models have been useful for the study of genotype-phenotype maps, where the specification of the building blocks can be viewed as a genotype and the resulting structure as

the respective phenotype [14, 18, 19]. This approach can be combined with genetic algorithms to model evolutionary processes [14]. Despite being a very abstract model, real biological self-assembly phenomena can be directly and meaningfully mapped to this Polyomino model. For example, the sickle-cell mutation of haemoglobin that leads to unbound protein aggregation can be modelled using Polyominoes [18]. Furthermore, experimental implementations of the Polyomino model have already been realised using DNA tiles [20].

The graph-based approach introduced in this paper aims to identify whether a given set of self-assembly building blocks will produce a final structure deterministically or nondeterministically, and whether that structure will be bound or unbound. The algorithm can be generalised to building blocks of any regular geometry and any number of binary attractive interactions. It therefore has potential applications in the study of protein complexes and protein aggregation, as well as bio-engineering and nanotechnology.

LATTICE SELF-ASSEMBLY

Following the model introduced in [17] a lattice self-assembly tile set consists of one or more tiles types. Each tile has interactive interfaces on some or all of its four faces, and different tile types exhibit different configurations of these interfaces. The interfaces, sometimes referred to as ‘colours’ [14, 17] can be represented as integers, and interactions between them can be described by a binary interaction matrix. The interactions can be symmetric, when an interface binds to another of the same type, (e.g. $1 \leftrightarrow 1, 2 \leftrightarrow 2, \dots$) or asymmetric, when it binds to an interface of a different type (e.g. $1 \leftrightarrow 2, 3 \leftrightarrow 4, \dots$), while 0 denotes a neutral interface. The interactions are infinite in strength, meaning that if two tiles come into contact, they will bind to each other irreversibly if the

two faces in contact have interactions that are compatible according to the interaction matrix. The assembly algorithm is fully stochastic. After the assembly is seeded with one random tile from the tile set, the assembly process undergoes the following steps: (a) a random face on the seed is chosen (b) a new tile is selected randomly from the tile set, (c) attachment to the selected face on the seed tile is attempted with a random orientation of the newly selected tile, and (d) if the interfaces on the faces in contact are compatible, the new tile is attached. Steps (a-d) are then repeated on a random face on the structure that is being formed until there are only neutral (interaction type 0) faces on the perimeter of the structure, in which case it cannot grow any further. Note that cooperative binding is not possible in our algorithm. Unless stated otherwise, we will assume asymmetric interactions for the remainder of this paper, such that each odd interface type $i = 1, 3, 5, \dots$ interacts with its $i + 1$ conjugate type, for example $1 \leftrightarrow 2, 3 \leftrightarrow 4, \dots$. See Figure 1 for an illustration of a simple self-assembly tile set.

DEFINITIONS OF STRUCTURE AND DETERMINISM

Definition of a tile structure: A structure is defined in terms of the relative positions and orientations of the building block instances, and the building block types. The definition of a structure is therefore independent of its absolute position or orientation as a whole on the lattice. Note that the seed tile is chosen randomly, which means that most structures will at least vary in terms of these degrees of freedom, even if they are otherwise deterministic.

Deterministic assembly tile set: An assembly tile set that always produces the same *structure* (in the limit $N \rightarrow \infty$ for unbound assemblies).

There are several variations of this definition of determinism that can be adopted within this lattice model. These are listed below. The first two are less strict than the above definition, and the last one is more restrictive.

Type- and orientation-independent determinism: Ignoring the tile types and orientations in the structure, the only criterion is that the same overall shape of the structure is produced by the assembly algorithm.

Orientation-independent determinism: Ignoring the tile orientations, the tile types and overall shape of the structure have to be the same in every assembly.

Absolute determinism: The structure is completely determined by the tile set, even with respect to its absolute position and orientation.

This paper focuses on the simplest definition of determinism, which is the first one given above, requiring tile types, orientations, and relative positions to be fixed, but leaving the absolute position and orientation unspecified, as the random seed would otherwise render most tile sets

nondeterministic.

ASSEMBLY GRAPH

The *assembly graph* of a tile set is a graphical representation of the interaction network between tiles with interactive faces. The connections in this network specify pairs of tile faces whose interfaces interact. Any two faces can at most have one connection between them. Figure 1 (left) shows an example of an assembly graph.

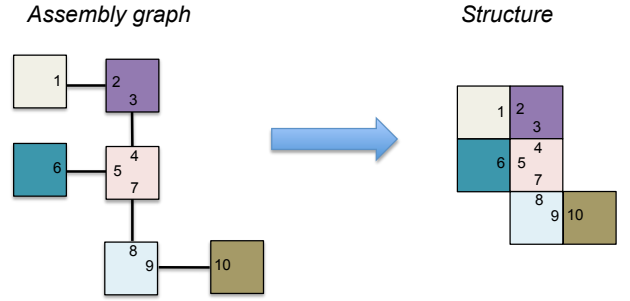


FIG. 1: Tree-like assembly graph: each colour appears only once in the tile set and no interaction loops exist between the tiles

A *branching point* in an assembly graph is defined as a tile face that connects to more than one other tile face. Branching points are the main source of nondeterminism because the stochastic choice of tiles during the assembly process means that one could choose one of several different tiles to successfully bind to a given face. However, branching points can also be deterministic. For instance, a tile with a single interactive face (SIF) can bind deterministically to several other tiles, e.g. $\{1, 0, 0, 0\} - \{2, 2, 2, 2\}$ has a branching point, but is a deterministic tile set. Branching points will be studied in greater detail below. An example of a structure with a branching point is presented in Figure 2.

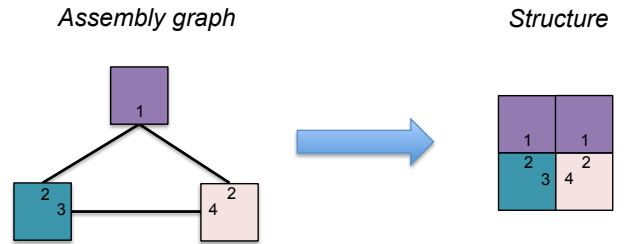


FIG. 2: Graph interaction network with a branching point and its translation into a structure via the self-assembly process

GRAPH JOINTNESS

We do not allow a graph to be disjoint, as this would lead to *seed dependence* [20], which means that a different final structure would emerge for different seed tiles.

TREE-LIKE ASSEMBLY GRAPHS WITHOUT BRANCHING POINTS ARE RULE-DETERMINISTIC

In a tree-like assembly graph without branching points, each tile face interacts only with a single other active face in the tile set, as shown in the example in Figure 1.

We define an *assembly graph walk* as a walk along the interaction graph, which consists of two alternating steps: Starting on a connected tile face the first step is to move along a connection in the graph to another tile face. The second step is to move to a different connected face on the same tile, before proceeding again with the first step. This walk mirrors the assembly process, as the first step represents the binding of two tiles, and the second step reflects the fact that after binding a face is unavailable to bind another tile.

Note that the assembly graph formalism does not allow the detection of steric effects, whereby a structure grows into itself. It therefore only allows us to determine whether a tile set is 'rule-deterministic', meaning deterministic according to its interaction rules. A rule-deterministic tile set can still be nondeterministic due to steric effects. This limitation will be discussed in more detail below.

Tile sets with tree-like assembly graphs that have no branching points are deterministic up to steric effects, because each active face can only bind to one other tile face, and each tile has a list of neighbours uniquely defined, up to steric effects. It should also be noted that, if the assembly graph is tree-like without branching points, there will always exist at least two tiles in the tile set with only one single interactive face (SIF tiles). This observation can be seen as a parallel to the theorem that every non-edgeless finite tree has two leaf nodes [21]. These properties of tree-like assembly graphs will be used in the analysis of more complex assembly graphs later on.

ALTERNATIVE TREE-LIKE ASSEMBLY PROCEDURES

There can be several assembly avenues, giving rise to identical overall lattice shape (think of the different ways a cross can be produced, see Figure 3). These assemblies will not code for the same *structure* as defined above (in terms of tile types, positions, and orientations) but will result in type- and orientation-independent determinism. See figure 3 for clarifications.

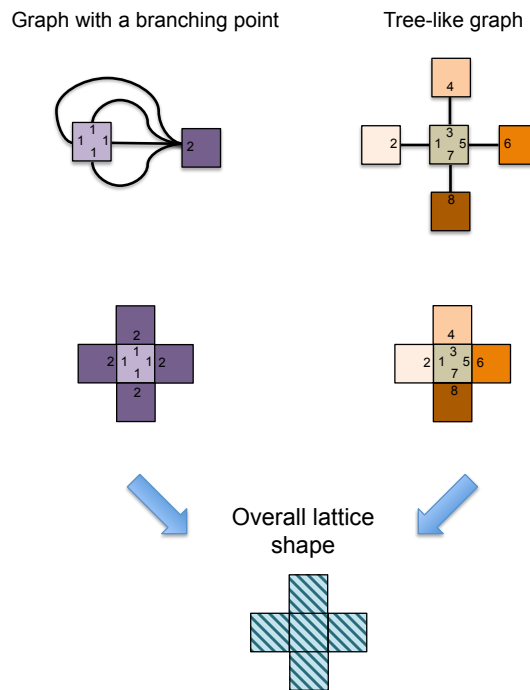


FIG. 3: Two alternative assembly procedures, producing a cross lattice shape. The right hand graph illustrates the alternative tree-like representation of the graph on the left

In general a branching point tile with a Single Interactive Face (SIF tile), whose colour g on the interactive interface does not appear anywhere else in the tile set possesses an alternative tree-like representation. To realise such alternative representation, introduce as many new SIF tiles in the tile set as the number of faces that colour g binds to, and connect them individually to each of the faces that the SIF tile branching point originally connects to.

We also note that if two interacting colours are both present on more than one interactive face each, then assembly is nondeterministic.

These alternative representations are important in the context of the next section which describes a way to recognise determinism from inspection of the self-assembly tile set.

SIF tiles elimination procedure

Consider a graph with a SIF tile connecting to several faces in the assembly. As shown in top image of figure 3, such a SIF branching point tile can be replaced by tree-like assembly avenues. Based on this we introduce the following procedure for eliminating SIF tiles in order to study the deterministic nature of a tile set:

SIF tiles elimination procedure: Transform SIF branching point tiles (whose colour appear only once in the whole tile set) into tree-like assembly representation

as outlined in the section above. Then, proceed with eliminating SIF tiles from the assembly tile set, turning into 0s the colours of the tiles the SIF tiles bind to. If the graph is tree-like and therefore deterministic, through the elimination procedure described above, a deterministic self-assembly tile set with no loops will always be reduced to two connected SIF tiles.

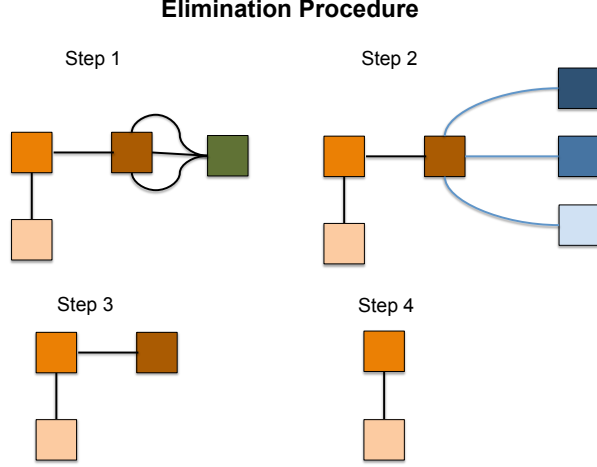


FIG. 4: Example of SIF elimination procedure on an example graph, recognising its deterministic nature. In step 2, the branching point is transformed into an equivalent tree-like assembly description

LOOP GRAPHS

Loop graph or subgraph: A loop is a set of tiles that give rise to a cyclic assembly graph walk, which means that, given a subset of n tiles from an assembly tile set of N tiles ($n \leq N$), if starting from any of the n tiles in the subset there exists one assembly graph walk that leads in n steps back to the first tile in the assembly subset with each tile being traversed once throughout the walk, then this set of tiles forms a loop graph ($n = N$) or loop sub-graph ($n < N$).

Loop rank observations

The rank of a loop defines the number of times each tile in the loop is used during the assembly of the structure. In a loop, after all tiles in the loop have been used exactly once during assembly, reattachment of the 1st tile can happen with an orientation shift by an angle $\theta = \frac{n\pi}{2}$, $n = 0, 1$ or 2 . Therefore, in two dimensions, there can be four types of loops:

Rank 1: in loops of rank 1, each tile is used exactly once. This means that seeding the assembly loop randomly from any tile in the self-assembly tile set, the last

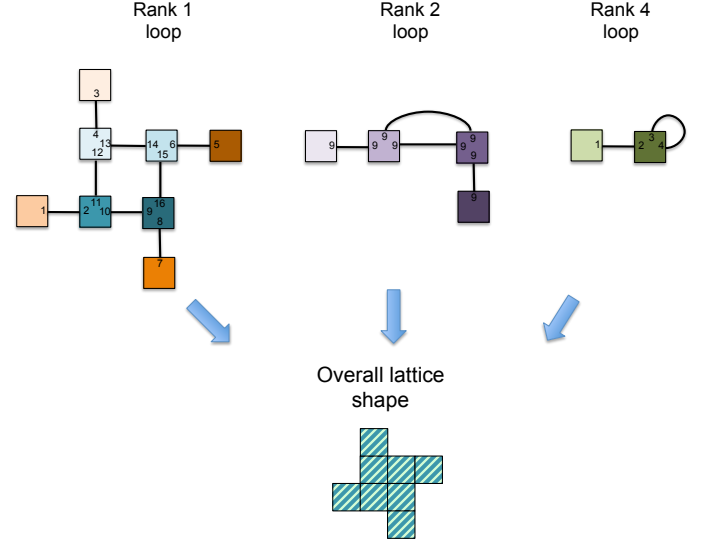


FIG. 5: Example of loops of rank 1,2 and 4 giving rise to the same structure

tile in the loop reattaches to the seed with matching tile faces.

Rank ∞ : the seed is to be reattached with no orientational shift, then the structure will keep growing forever, such as the single self-assembly tile $\{1, 0, 2, 0\}$.

Rank 2: if the seed tile is reattached with $\theta = \pi$, each tile will appear altogether twice before the structure reattaches to the initial seed tile location, implying that the loop is of rank 2.

Rank 4: Similar to the argument for rank 2 loops, if $\theta = \pm \frac{\pi}{2}$, the loop is of rank 4.

Loop rank examples can be found in figure 5.

Establishing boundedness

If there are at least 2 loops of rank higher than 1, growth becomes unbound.

As an example, consider the two loops shown in figure 6: loop (F, B, E) and loop (A, B, C, D) have rank 4. Each time an assembly of loop (F, B, E) is performed, tile B will be assembled four times in the structure. Loop (A, B, C, D) in turn, will create copies of tile B another 4 times, triggering assembly of loop (F, B, E) and so on. Hence the two loops keep regenerating each other.

Eliminating loops of rank 1

Loops of rank 1 possess alternative tree-like assembly avenues, in a similar fashion to SIF branching point tiles as described above. For this reason, we can directly turn loops of rank 1 into their alternative tree-like description by eliminating one of the edges in the loop before the SIF

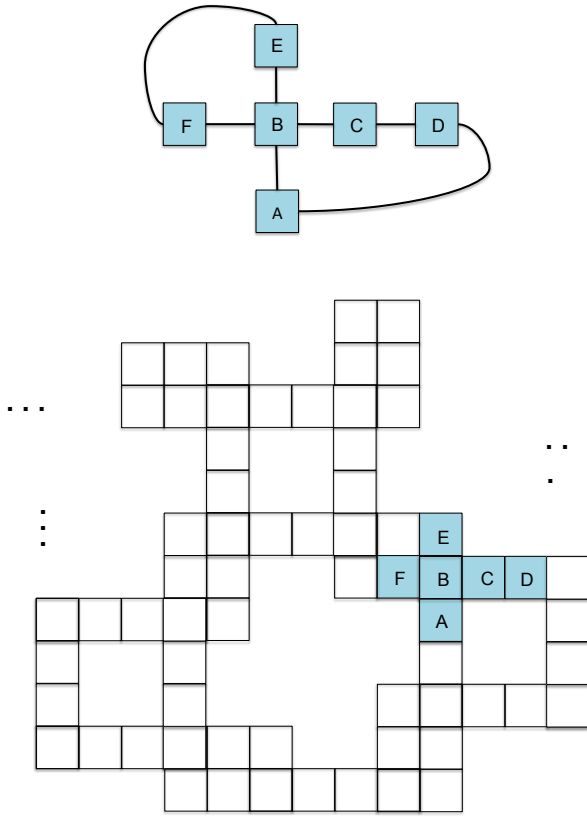


FIG. 6: Unbound structure made of two regular loops of rank 4. Each time loop (F, B, E) is assembled, 4 new (A, B, C, D) loops are produced, which in turn generate 4 (F, B, E) loops, producing unbound growth

elimination procedure. See figure 7 for an example.

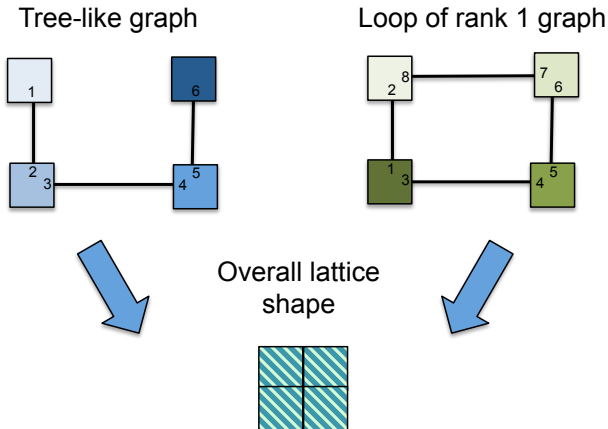


FIG. 7: Minimal and Maximal assembly procedures producing a square lattice shape

If after SIF elimination, loops of rank higher than 1 are present along with irreducible branching points, assembly is nondeterministic. An example of this can be found in figure 8. Branching points would have to be traversed

stochastically during assembly and neighbours to a tile face would not be uniquely defined.

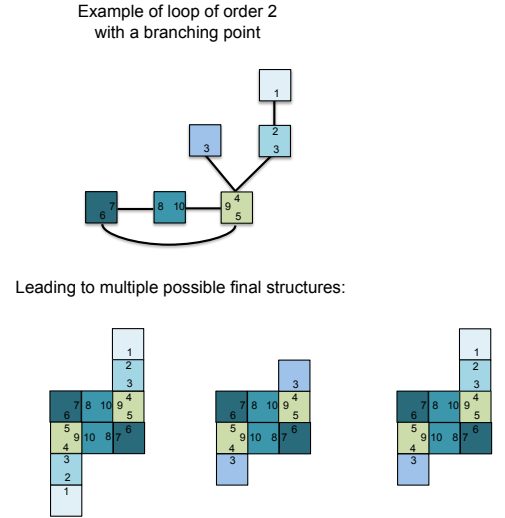


FIG. 8: Example of nondeterministic tile graph because of the presence of a loop of rank 2 and a branching point. It is not possible to design an equivalent tree-like description of this graph because the presence of a loop of rank 2 implies that the branching point will have to be traversed stochastically during growth

SYMMETRY

So far, we have disregarded symmetric tiles because they produce branching points of a different nature from the branching points analysed above and more care has to be taken. In fact, we can get rid of branching points resulting from geometrical symmetry in the tiles via the introduction of alternative assembly descriptions. Consider for instance the tile set $\{1, 1, 1, 1\} - \{2, 2, 2, 2\}$ which produces unbound deterministic growth. We can introduce alternative deterministic assembly pathways getting rid of symmetry explicitly, yet preserving the same growth behaviour, as shown in the example in figure 9. Full details on the symmetrisation procedure are included in the appendix.

STERIC EFFECTS

In the preceding sections we have addressed the question whether the assembly interactions alone make a tile set deterministic or non-deterministic, and bound or unbound. However, steric effects can also affect the growth of structures and produce sterically bound structures that are unbound according to the interaction rules or sterically nondeterministic structures that are deterministic according to the rules. In order to recognise steric

SYMMETRY REMOVAL EXAMPLE

Tile set with symmetric tiles: $\{1,1,1,1\}-\{2,2,2,2\}$

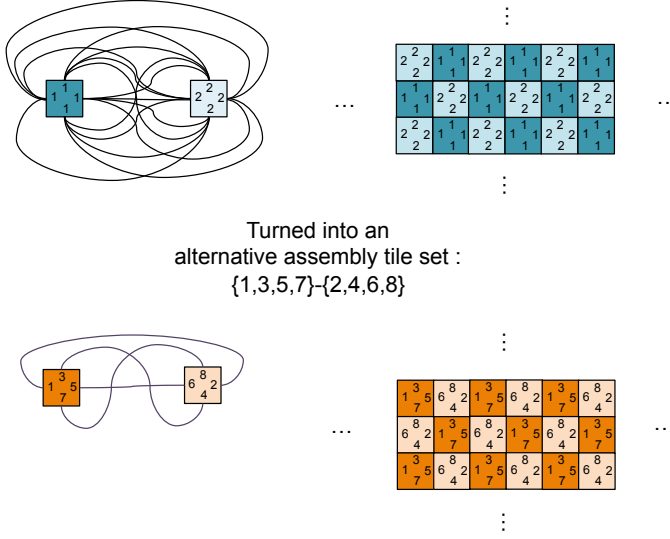


FIG. 9: The symmetry removal procedure gets rid of branching points that arise due to symmetry in the tile geometry, while producing the same growth behaviour as the original self-assembly tile set.

effects, we can grow the structure a single time: if any coloured faces are mismatched during growth, and if the complementary colour of at least one of these appears elsewhere in the assembly set, then some assembly avenues are sterically prevented. See Figure 10. Therefore, once rule-determinism has been detected, growing the structure once allows for the detection of steric effects.

Example of steric nondeterminism with tile set $\{1,0,0,0\}-\{2,0,3,0\}-\{5,4,0,0\}-\{7,6,0,0\}-\{9,8,0,0\}-\{10,0,0,0\}$

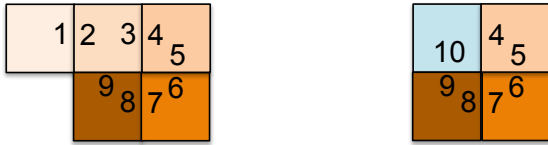


FIG. 10: Assembly tile set $\{1,0,0,0\}-\{2,0,3,0\}-\{5,4,0,0\}-\{7,6,0,0\}-\{9,8,0,0\}-\{10,0,0,0\}$ is rule-deterministic. However, growing the structure directly can lead to two possible final structures because of steric effects. Building the structure once results in mismatched active faces as shown in the figure (colour 9 is mismatched on the left structure and colour 4 on the right structure), allowing us to detect steric nondeterminism

FINAL PROCEDURE

The flowchart in figure 11 illustrates all of the steps to be implemented when determining the (non-)deterministic (un-)bound nature of a self-assembly tile set.

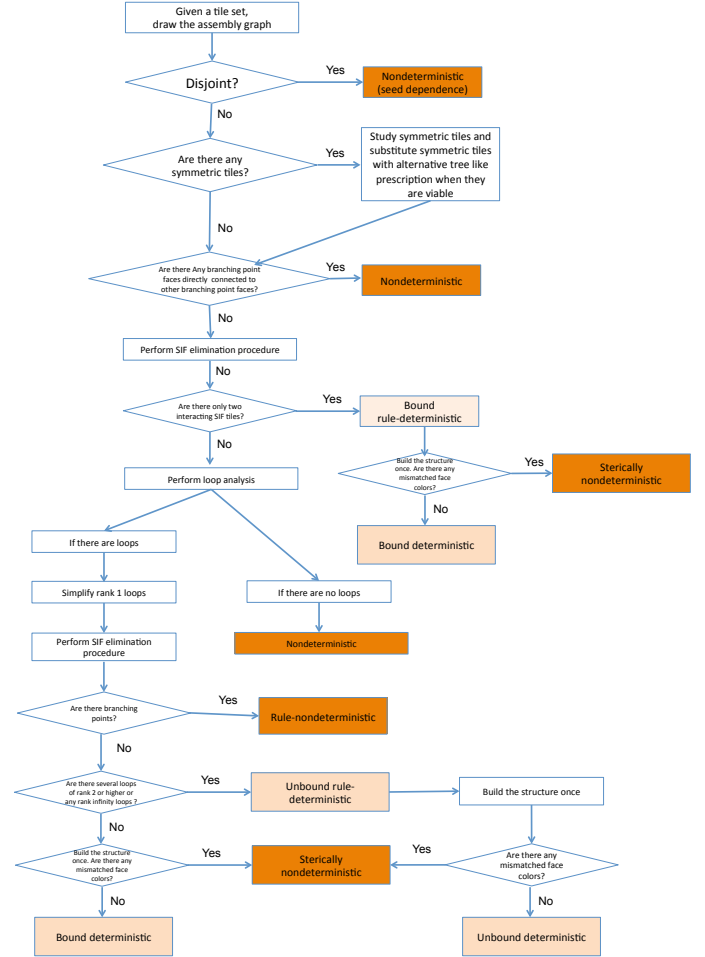


FIG. 11: Final procedure diagram

DISCUSSION

Other dimensions and geometries

The algorithms presented in this paper for recognising tile self-assembly determinism can be directly extended to other regular geometries. In extending the determinism rules, the loop rank theorems need to be adjusted, taking into account the appropriate θ -shifts introduced during structure assembly. The plane can regularly be tiled using triangles, squares and hexagons among the regular polygons [22].

The same theorems will apply in the reduction procedure, and one will simply have to consider that for triangles

there can only be loops of finite rank 1, 2 and 6 while for hexagons, there can be loops of finite rank 1,2,3,4 and 6. Some loop examples for regular hexagons are presented in figure 12.

Similarly, an extension of the algorithms can be performed in 3 dimensions. For instance for cubes, the finite loop ranks can still be only 1,2 and 4, as for squares because the possible orientational shifts can only be $\frac{n\pi}{2}$ with $n = 0, 1, 2$ or 3. Therefore, we recognise that the deterministic rules derived in this paper can be extended to higher dimensions and other regular geometries.

Hexagonal Lattice

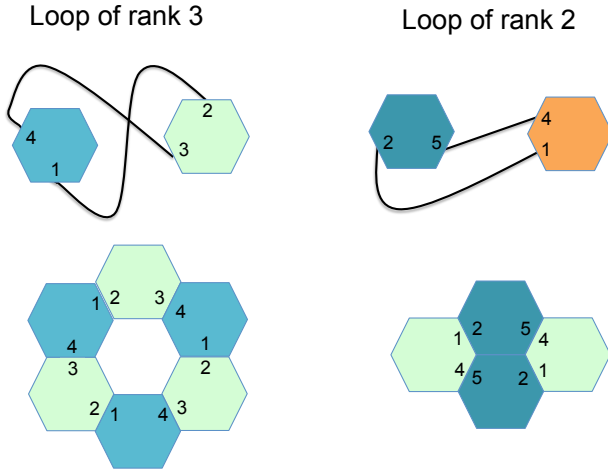


FIG. 12: Regular Loops of rank 3 and 2 on an hexagonal lattice

Non-deterministic genotype-phenotype maps

In [15], we studied in detail nondeterministic self-assembly of single lattice tiles and mixtures of two tiles at varying relative concentrations with colours 0, 1 and 2, where 1 interacts with 2, and 0 is a neutral colour. We were able to classify tile sets in terms of their behaviour upon variation of tile density, recognising critical transitions from bound to unbound growth behaviours and non critical density transitions. In total, 106 topologically distinct tile sets were identified. Tile graph methods now allow us to determine with no need of direct assembly the (un-)bound and (non-)deterministic tile set behaviour, completing the picture. In figure 13 we present the bound deterministic and unbound deterministic tile sets with two tiles and 3 colours (0, 1 and 2) under asymmetric interactions. The number of deterministic tile sets is much smaller than the nondeterministic ones, 10 versus 106.

Furthermore, we also provide a table with all 106 tile sets and their corresponding tile graphs in the supplementary

material for symmetric and asymmetric interactions.

2-tile sets with asymmetric interactions and 0, 1 and 2 colours

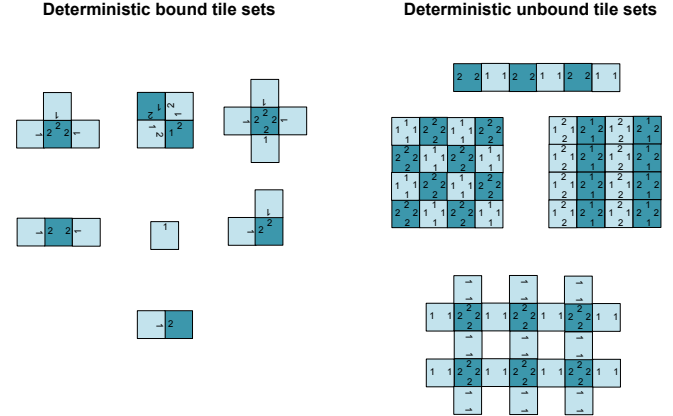


FIG. 13: All deterministic assembly tile sets with two tiles and two active colours

Genotype-phenotype maps with 2 tiles and 8 colours

The lattice self-assembly model described here has been used as an abstract model of the genotype-phenotype map of protein quaternary structure [18]. In these models the interactions of the building block set constitute the genotype and the assembled structure represents the phenotype. This work focused on deterministic assembly, while non-deterministic building block sets were regarded as a single unfavourable phenotype, and largely ignored. The assembly graph framework allows us to define the assembly graphs as the phenotypes, and thus explore the genotype-phenotype map of non-deterministic tile sets. As non-deterministic self-assembly in the form of protein aggregation is a hallmark of numerous diseases, the study of such non-deterministic genotype-phenotype maps can help us to better understand these phenomena. Here we consider the assembly graph genotype-phenotype (AGGP) for two tiles and eight colours. Figure 14 shows the frequency of a phenotype versus the rank of its frequency within the map, which reveals a similar heavy-tailed distribution to those described in [18]. In contrast the Polyomino GP map of deterministic lattice self-assembly, we are now accounting for nondeterministic structures in our AGGP map as well.

The total number of topologically distinct assembly graphs with two tiles and eight colours is 1136. Of these, twelve yield bound deterministic structures, which make up 5876685 of all possible 1.6×10^7 degenerate genotypes, or 36.7% of all genotype space. In [14], the same genotype space has been studied, focusing on the Genotype-Phenotype map between the self-assembly tile set and

the final assembled structure. In this study, 38.9% of all genotypes were classified as bound and deterministic, through repeated direct assembly of the phenotype from the same seed. The number of bound deterministic genotypes is lower for the AGGP map because steric determinism is considered as a form of non-deterministic behaviour, whereas the GP map in [14] classifies self-assembly tile sets as deterministic irrespective of mismatches between tile face colours, provided that multiple assemblies of the same self-assembly tile set yield the same final structure. The discrepancy in number of bound deterministic phenotypes is also caused by the seeding choice, which is stochastic for the self-assembly tile sets studied in the AGGP map, and specified for the models explored in [14].

In addition to general genotype phenotype maps, the assembly graph framework can also be used to illustrate the function of the sickle cell mutant as modelled using the Polyomino model in [18]. In this representation the α and β chains of wild-type haemoglobin Hb A can be mapped to the $\{1, 3, 0, 0\} - \{2, 5, 0, 4\}$ tile set, and the sickle-cell mutant Hb S can be represented by the $\{1, 3, 0, 0\} - \{2, 5, 6, 4\}$. The point mutation through the introduction of colour ‘6’ results in unbound deterministic growth giving rise to the sickle cell anaemia disease. Using the assembly graph framework we can see that tile set $\{1, 3, 0, 0\} - \{2, 5, 0, 4\}$ possesses a loop of rank 2 and that the mutation of the second tile, which results in the tile set $\{1, 3, 0, 0\} - \{2, 5, 6, 4\}$, leads to the introduction of a second loop into the assembly graph, of rank 4. We can take this further, noting that biological systems that are assembled through loops of rank higher than 1 are prone to this type of mutations to unbound growth via the introduction of a further single loop of rank higher than 1 that is added.

We also suggest that protein misfolding and resulting changes of interface angles in quaternary structures can be recast in these terms, as a loop of rank infinity that is introduced or a loop of rank higher than 1 that is added.

Self-limiting cluster growth and limitations

In previous experimental work [20], we have compared single-seed and multi-seed self-assembly, observing that asymmetric interactions can lead to self-limiting growth in multi-seed assembly through local steric effects. The graph methods developed here cannot predict *a priori* steric effects and therefore can never predict whether an assembly tile set will run into self-limiting growth because of such geometrical constraints.

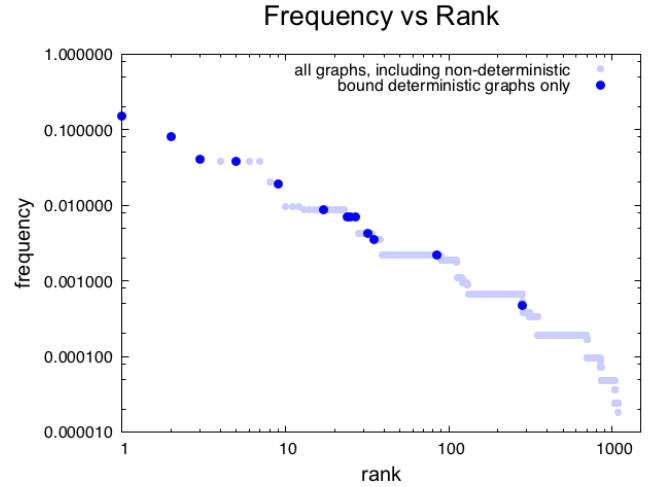


FIG. 14: Phenotypic frequency vs phenotypic frequency rank in the AGGP map for the space of two tiles and eight colours, showing both deterministic (dark blue) and nondeterministic (light blue) phenotypes

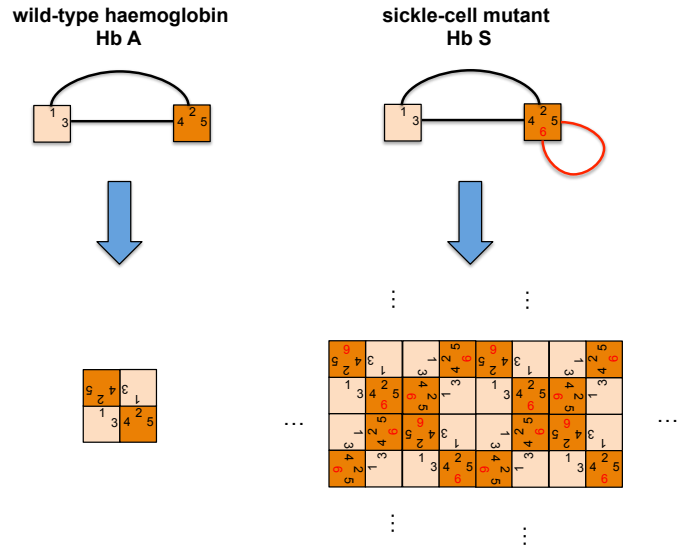


FIG. 15: The wild-type and sickle-cell mutant of the haemoglobin complex can be modelled using Polyominoes [18]. Here we show that this description can be extended to a graph representation, which shows that the sickle cell mutation introduces a new loop of rank 4 in the tile set. Coupled with the loop of rank 2 present in the wild type Hb A this results in unbound growth.

CONCLUSION

With the assembly graph framework we have introduced a new approach for the study of self-assembling systems that can help us to determine whether self-assembly outcomes are bound or unbound, and deterministic or non-deterministic.

This also offers a powerful tool for the study of Geno-

type Phenotype maps, by expanding the exploration of phenotypes to the non-deterministic realm, which offers a very general model of dysfunction and disease in the context of biological self-assembly.

Appendix: Symmetry removal

Symmetric tiles in an assembly self-assembly tile set can be substituted with alternative assembly pathways (see Figure 9) eliminating branching points which arise because of tile symmetry and which do not lead to nondeterministic structure growth. Here we present a detailed description for each possible combination of symmetric tiles appearing in the self-assembly tile set.

Symmetric tiles can only be of type $\{A, A, A, A\}$, $\{A, C, A, C\}$, $\{A, B, A, B\}$ (where A interacts with B , C with D) and $\{A, E', A, E'\}$, indicating with E' any other colour that does not interact with any of the A, B, C, D colours.

Observe that having tile $\{A, B, A, B\}$ in the tile set either leads to seed dependence if no other A or B colours are present on other interacting tiles the tile set, or it leads to nondeterminism if any A or B colours appear on other tiles. For this reason, the possibility of having a $\{A, B, A, B\}$ tile is excluded from this discussion as it should never be taken into account in the symmetrisation procedure.

Similarly, the presence of tiles $\{A, C, A, C\}$ and $\{B, D, B, D\}$ leads to seed dependence or nondeterminism if there are any other tiles in the tile set. Nevertheless, the single tile made of $\{A, B, A, B\}$ alone and tile set $\{A, C, A, C\}$ - $\{B, D, B, D\}$ are unbound and deterministic. We exclude the presence of these cases in an assembly tile set from the remainder of our symmetrisation procedure.

Then, defining $*$ as any colour which does not interact with A, B, C or D , there exist only 5 possible tile geometries containing a B face: $\{B, *, *, *\}$, $\{B, B, *, *\}$, $\{B, B, B, *\}$, $\{B, *, B, *\}$, $\{B, B, B, B\}$, if we ignore tiles containing both B and A colours at the same time as they lead to nondeterminism directly when interacting with any of the $\{A, A, A, A\}$, $\{A, C, A, C\}$ tiles.

Only a choice of either of the $\{A, A, A, A\}$ or $\{A, C, A, C\}$ tiles, and any of the $\{B, *, *, *\}$, $\{B, B, B, B\}$ or $\{B, E', B, E'\}$ tiles is deterministic (colour E' has been chosen without loss of generality as a colour not interacting with A, B, C, D which can also be a neutral colour).

Furthermore, tile $\{A, E', A, E'\}$ does not need to be considered because it would lead to redundant cases, whether E' interacts or not with any of the $*$ colours. All cases with colours A and B swapped have also been ignored because they would lead to redundant combinations.

Our aim is to convert branching points deriving from symmetric interactions into alternative deterministic assembly descriptions, introducing new tiles and colours replicating the behaviour of the original sets.

Therefore, perform the following transformations, which are graphically presented in Figure 16:

- 1) Convert the pair $\{B, B, B, B\}$ - $\{A, C, A, C\}$ into 3 new tiles displaying colours $\{B1, B2, B3, B4\}$ - $\{A1, C, A3, C'\}$ - $\{A2, C, A4, C'\}$, where colour $B1$ interacts with colour $A1$, $A2$ with $B2$ and so on.
- 2) Similarly, $\{B, B, B, B\}$ - $\{A, A, A, A\}$ becomes $\{B1, B3, B2, B4\}$ - $\{A1, A3, A2, A4\}$.
- 3) $\{B, B, B, B\}$ - $\{A, *, *, *\}$ and $\{B, *, *, *\}$ - $\{A, C, A, C\}$ remain unchanged because they are deterministic branching points and get reduced directly in the SIF elimination procedure.
- 4) $\{B, E', B, E'\}$ - $\{A, C, A, C\}$ becomes $\{B2, E', B1, E'\}$ - $\{A1, C, A2, C'\}$.
- 5) In the presence of two or more symmetric tile cases, Steps 1) – 4) need to be combined to remove branching point resulting from symmetric tiles. For instance, in tile set $\{A, C, A, C\}$ - $\{D, D, D, D\}$ - $\{B, E', B, E'\}$, step 1 and 4 have to be combined, and the symmetrisation procedure results in $\{B1, E', B2, E'\}$ - $\{A2, C1, A1, C3\}$ - $\{B3, E', B4, E'\}$ - $\{A4, C2, A3, C4\}$ - $\{D1, D2, D3, D4\}$.

After reducing branching points resulting from tile symmetries leading to deterministic self-assembly, the assembly tile sets can be studied through the procedure described in the main text to assess its bound deterministic nature.

[1] Erik Winfree, Furong Liu, Lisa A Wenzler, and Nadrian C Seeman. Design and self-assembly of two-dimensional DNA crystals. *Nature*, 394(6693):539–544, January 1998.

SYMMETRY REMOVAL PROCEDURE

Graphs with branching points in tile sets with symmetric tiles

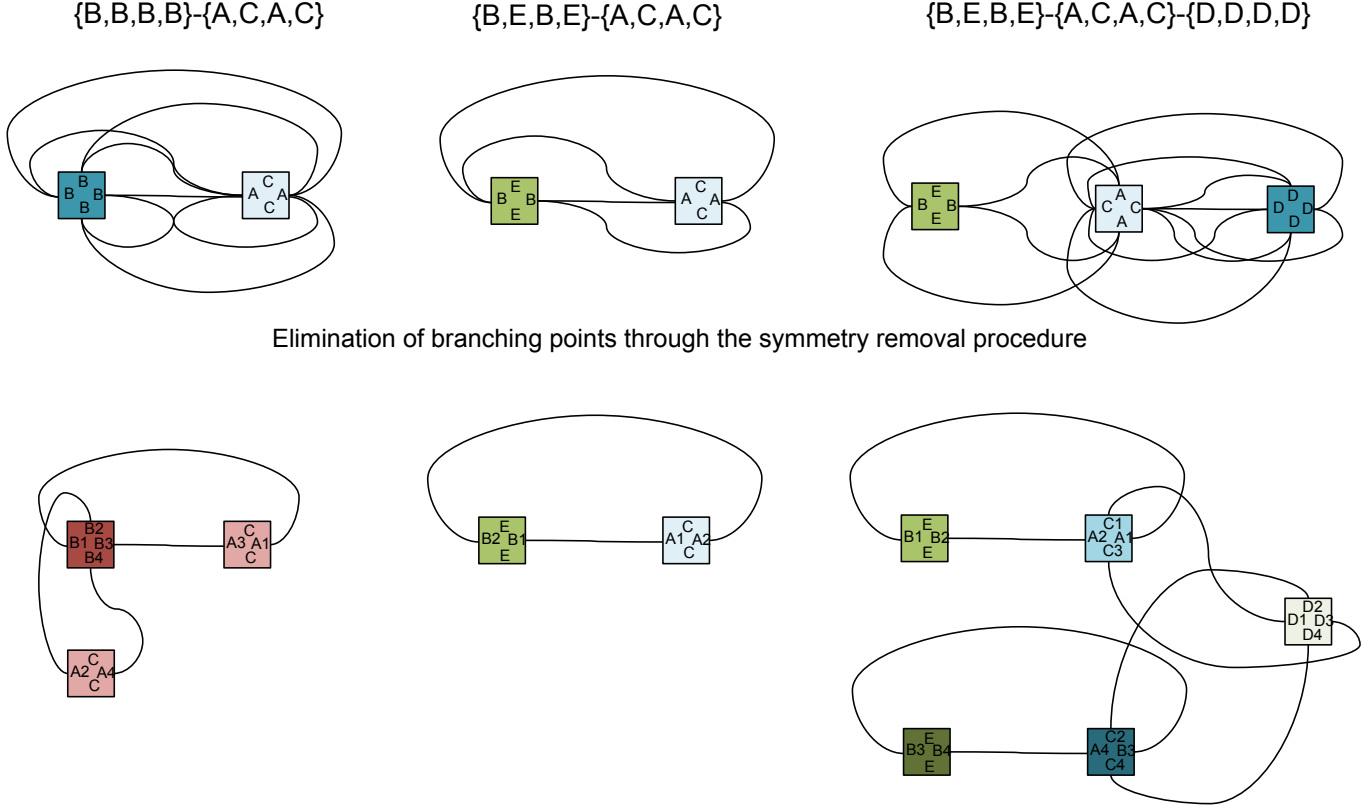


FIG. 16: These examples show how to transform assembly tile sets with symmetric tiles to alternative assembly descriptions, preserving the same growth behaviour but eliminating branching points which simply arise from the presence of symmetric tiles in the tile set

- [2] C. Mao, T. H. LaBean, J. H. Reif, and N. C. Seeman. *Nature*, 407(6803):493–496, 2000.
- [3] A. Chworos, I. Severcan, A. Y. Koyfman, P. Weinkam, E. Oroudjev, H. G. Hansma, and L. Jaeger. *Science*, 306(5704):2068–2072, 2004.
- [4] R. P. Goodman, I. A. T. Schaap, C. F. Tardin, C. M. Erben, R. M. Berry, C. F. Schmidt, and A. J. Turberfield. *Science*, 310(5754):1661–1665, 2005.
- [5] P. W. K. Rothmund. *Nature*, 440(7082):297–302, 2006.
- [6] K. Fujibayashi, R. Hariadi, S. H. Park, E. Winfree, and S. Murata. *Nano Letters*, 8(7):1791–1797, 2007.
- [7] Y. Ke, L. L. Ong, W. M. Shih, and P. Yin. *Science*, 338(6111):1177–1183, 2012.
- [8] E. D. Levy, J. B. Pereira-Leal, C. Chothia, and S. A. Teichmann. *PloS Computational Biology*, 2(11):e155, November 2006.
- [9] G. Villar, A. W. Wilber, A. J. Williamson, P. Thiara, J. P. K. Doye, A. A. Louis, M. N. Jochum, A. C. F. Lewis, and E. D. Levy. *Physical Review letters*, 102(11):118106, 2009.
- [10] S. I. A. Cohen, S. Linse, L. M. Luheshi, E. Hellstrand, D. A. White, L. Rajah, D. E. Otzen, M. Vendruscolo, C. M. Dobson, and T. P. J. Knowles. *Proceedings Of The National Academy Of Sciences Of The United States Of America*, 110(24):9758–9763, 2013.
- [11] A. Zlotnick. *Journal of molecular biology*, 241(1):59–67, 1994.
- [12] Jacob Israelachvili. Self-assembly in two dimensions: surface micelles and domain formation in monolayers. *Langmuir*, 10(10):3774–3781, 1994.
- [13] G. Krausch and R. Magerle. *Advanced Materials*, 14(21):1579–1583, 2002.
- [14] I. G. Johnston, S. E. Ahnert, J. P. K. Doye, and A. A. Louis. *Phys. Rev. E*, 83:066105, Jun 2011.
- [15] S. Tesoro and S. E. Ahnert. Nondeterministic self-assembly of two tile types on a lattice. *Phys. Rev. E*, 93:042412, Apr 2016.
- [16] H F Bunn. Pathogenesis and treatment of sickle cell disease. *The New England Journal of Medicine*, 337(11):762–769, September 1997.

- [17] S. E. Ahnert, I. G. Johnston, T. M. A. Fink, J. P. K. Doye, and A. A. Louis. *Phys. Rev. E*, 82:026117, Aug 2010.
- [18] Sam F Greenbury, Iain G Johnston, Ard A Louis, and Sebastian E Ahnert. A tractable genotype–phenotype map modelling the self-assembly of protein quaternary structure. *Journal of The Royal Society Interface*, 11(95):20140249, 2014.
- [19] Sam F Greenbury, Steffen Schaper, Sebastian E Ahnert, and Ard A Louis. Genetic correlations greatly increase mutational robustness and can both reduce and enhance evolvability. *PLoS Comput Biol*, 12(3):e1004773, 2016.
- [20] S. Tesoro, K. Göpfrich, T. Kartanas, U. F. Keyser, and S. E. Ahnert. Nondeterministic self-assembly with asymmetric interactions. *Phys. Rev. E*, 94:022404, Aug 2016.
- [21] from MathWorld-A Wolfram Web Resource E. W. Weisstein. Tree leaf, June 2016.
- [22] Branko Grünbaum and Geoffrey C Shephard. Tilings by regular polygons. *Mathematics Magazine*, 50(5):227–247, 1977.